

Homework Assignment #3

- 1 Read the following code, and write comment in the code to mark the point in the program's execution that constructors of the Student class is called, and which constructor is called. Also write comment next to the line of code where desctructor is called.

```
class Student {
public:
    friend Student OutputDuplicate (Student obj);

    Student (); //default constructor
    Student (string n);
    Student (Student obj); //copy constructor
    ...
private:
    string name;
    ...
};

Student OutputDuplicate (Student obj)
{
    cout <<"Student record:";
    cout <<obj.name << endl;

    Student * anotherObj = new Student ("Jane Smith");
    *anotherObj = obj;

    return *anotherObj;
}

int main()
{
    Student a, b ("John Smith");

    b = OutputDuplicate (a);
}
```

- 2 Answer the following questions by reviewing the textbook (using the index at the end of the book to look up where the terms are discussed in the book).
 - (a) List the three usage of modifier *const* in C++ and give the examples.

(b) List at least two usage of modifier *static* in C++ and give the examples.

(c) List at least three different directives (lines in C++ code that starts with `#`), and explain what each of them do? Which stage of the C++ compiler process directives?

3 When you define a class, and do not provide any constructor, then the C++ provided no-parameter constructor will be used. Please type in the following code, add necessary header files, compile and run the code, and answer the question based on your observation.

```
class YourClassName
{
public:
    friend ostream & operator<< (ostream & outs, const YourClassName & o)
    {
        outs << o.value << endl;
        outs << o.dvalue << endl;
        for (int i=0;i<10;i++)
```

```

        outs <<"a["<<i <<"]:" << o.a[i]<<endl;
        outs << o.c << endl;
        return outs;
    }
private:
    int value;
    double dvalue;
    int a[10];
    char c;
};

```

```

YourClassName a; // This is fine, as the system provides a default implementation for
// default constructor
cout << a << endl;
// Todo: How does the default constructor initialize the member variable for us?
// int member variable:
// double member variable:
// array variable
// char variable:
//

```

- 4 (Public/private/protected inheritance) Comment on compilation errors in the following code (without typing it up and run the compiler on it). Briefly explain why.

```

class base{
public:
    int get_b();
    int set_b(int bValue);
    int get_c();
    int set_c(int cValue);

    int a;
protected:
    int b;
private:
    int c;
};

class derived1: base{
public:
    void output(){
        cout <<a<<endl;
        cout <<b<<endl;
        cout <<c<<endl;
    }

private:
    char d;
};

class derived2:private derived{

```

```

public:
void output(){
cout <<a<<endl;
cout <<b<<endl;
cout <<c<<endl
cout <<d<<endl;
}
private:
    bool e;

};

int main(){
    base obj;

    cout <<obj.a<<endl;
    cout <<obj.b<<endl;

    derived1 obj1;
    cout <<obj1.a<<endl;
    cout <<obj1.b<<endl;

    derived2 obj2;
    cout <<obj2.a<<endl;
    cout <<obj2.b<<endl;
    cout <<obj2.c<<endl;

}

```

- 5 What's the slicing problem? Give an example that's different from the one given in the textbook or class. Can you store a collection of Employee objects (some are HourlyEmployees, some one SalariedEmployee) in an array of Employee objects? If not, what should you do?

- 6 (Extra credits: from Stackoverflow) A puzzle about constructor/destructor. Given the following class *One*, which you cannot change, you need to make the program that display a sentence: "One Two Three Four Five Six Seven". Hint: you need to make the constructor and destructor of each class output some word. But you are constrained as follows: every class cannot display more than one word in its constructor and destructor.

```
class One: public Two
public:
    One()
    {
    Three three;
    cout <<"something"; //change the word being cout here
    }
    private:
        Four _four;
};

int main()
{
    One one; // <-- This has to be here
    cout <<endl;

}
```